Available online at www.sciencedirect.com
SCIENCE DIRECT
Discrete Applied Mathematics 154 (2006) 912–929
DISCRETE APPLIED MATHEMATICS
www.elsevier.com/locate/dam

On constant composition codes

Wensong Chu[a], Charles J. Colbourn[a], Peter Dukes[b]

[a] *Department of Computer Science and Engineering, Arizona State University, Tempe, AZ 85287-5406, USA*
[b] *Department of Mathematics and Statistics, University of Victoria, Victoria, BC, Canada V8W 3P4*

Received 7 November 2003; received in revised form 14 June 2005; accepted 27 September 2005
Available online 8 November 2005

Abstract

A constant composition code over a k-ary alphabet has the property that the numbers of occurrences of the k symbols within a codeword is the same for each codeword. These specialize to constant weight codes in the binary case, and permutation codes in the case that each symbol occurs exactly once. Constant composition codes arise in powerline communication and balanced scheduling, and are used in the construction of permutation codes. In this paper, direct and recursive methods are developed for the construction of constant composition codes.

Keywords: Constant composition code; Constant weight code; Permutation code

1. Introduction

Communication over an electric power line has become an attractive alternative to cable television or telephone as a solution to the "last mile" problem of delivering information services to and within a home (see [18]). Modulation of the frequency can be used to accomplish this information encoding, but this causes a variation in power delivered on the line. If k different frequencies can be chosen, each information unit can be encoded as a codeword over the k-ary alphabet of frequencies [6,8]. Then the frequencies are transmitted sequentially, decoded to determine the information, and the power output remains as available electrical power. Without careful selection of codewords, power output on the line is not constant, and the information delivery interferes with the power delivery. Constant composition codes provide an acceptable solution, by allowing local variations that are small but ensuring that upon completion of each codeword, the power expended is the same for each information unit encoded.

More generally, constant composition codes arise in frequency hopping (FH), when a schedule is needed to determine frequencies on which to transmit; see [9]. When each frequency is to be used a specified number of times within a frame, each FH sequence is a codeword of constant composition. Indeed, whenever a different cost is associated with each symbol in the underlying alphabet, uniform cost of codewords leads to constant composition.

Two special cases, (binary) constant weight codes and permutation codes, have been studied in some depth; in this paper, we consider the generalization to constant composition codes.

E-mail addresses: Charles.Colbourn@asu.edu (C.J. Colbourn), dukes@math.uvic.ca (P. Dukes).

0166-218X/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2005.09.009

Let C be a k-ary code of length n and distance d on the alphabet $\{1, \ldots, k\}$. As usual, the elements of C are *codewords*, and the collection of alphabet symbols in the ith position of every codeword is the ith *column* of C.

Code C has *constant weight composition* $[n_1, \ldots, n_k]$ if every codeword has n_i occurrences of symbol i for $i = 1, \ldots, k$. (Since the alphabet is immaterial to a code, we may view the composition $[n_1, \ldots, n_k]$ as an unordered multiset, and not restrict the alphabet to $\{1, \ldots, k\}$.) Code C is a *constant composition code*, or CCC$([n_1, \ldots, n_k], d)$, or simply a CCC. Let $A([n_1, \ldots, n_k], d)$ denote the maximum size of such a CCC. Unless stated otherwise, we assume $n = \sum n_i$.

We say $[n_1, \ldots, n_k]$ is a *refinement* of $[m_1, \ldots, m_h]$ if there is a partition $\{I_1, \ldots, I_h\}$ of $\{1, \ldots, k\}$ such that $\sum_{i \in I_j} n_i = m_j$ for each j. In this case, we write $[n_1, \ldots, n_k] \preccurlyeq [m_1, \ldots, m_h]$. The *dual* of $[n_1, \ldots, n_k]$, written as $[n_1, \ldots, n_k]^*$, is the partition of n whose ith part is the number of n_j, $j = 1, \ldots, k$, which are greater than or equal to i.

When writing compositions, the exponential notation $n_1^{t_1} n_2^{t_2} \cdots n_h^{t_h}$ may be used to abbreviate

$$[\overbrace{n_1, \ldots, n_1}^{t_1}, \overbrace{n_2, \ldots, n_2}^{t_2}, \ldots, \overbrace{n_h, \ldots, n_h}^{t_h}].$$

In case the n_i are themselves exponents, we revert to the composition list to avoid confusion.

The following are easy consequences of the definitions, and generalize the results in [21] for ternary codes.

Lemma 1.1. (1) *If* $[n_1, \ldots, n_k] \preccurlyeq [m_1, \ldots, m_h]$, *then* $A([n_1, \ldots, n_k], d) \geqslant A([m_1, \ldots, m_h], d)$.

(2) *If* $d = d_1 + \cdots + d_k$, *then* $A([n_1, \ldots, n_k]^*, d) \geqslant \min_i A(1^{n_i}, d_i)$.

(3) *If* $d > 2(n - n_k)$, *then* $A([n_1, \ldots, n_k], d) = 1$.

(4) $A([n_1, \ldots, n_k], 2(n - n_k)) = \lfloor n/(n - n_k) \rfloor$.

(5) $A([2, n_1, \ldots, n_k], d) \geqslant \frac{1}{2} A([1, 1, n_1, \ldots, n_k], d + 1)$.

Only (5) requires an explanation. In a code with composition $[1, 1, n_1, \ldots, n_k]$, let 1 and 2 be the symbols in the first two parts. Without loss of generality, at least half of the codewords contain the 1 in a position prior to that of the 2. In this set of codewords, identify symbols 1 and 2. By selecting codewords in this way, the distance can drop by at most 1. Similar identifications are possible, but this appears to be the most useful example that is trivial.

Codes with constant composition 1^n are also known as *permutation arrays*, denoted by PA(n, d), and have been studied recently in [4]. Several of the constructions to follow for CCCs have been used for PAs. Codes with constant composition $[w, n - w]$ are the much-investigated constant weight binary codes [2].

For distance two, the largest code is easily determined. Indeed, $A([n_1, \ldots, n_k], 2) = \binom{n}{n_1, n_2, \ldots, n_k}$, the multinomial coefficient. Even for distance three, however, no general result for CCCs appears to be known, despite the fact that for permutation arrays the maximum can be achieved by the set of all even permutations. In this vein, we provide one minor result for illustrative purposes.

Lemma 1.2. $A(2^1 1^{n-2}, 3) \geqslant \lfloor n/2 \rfloor \cdot \lceil n/2 \rceil \cdot (n - 2)!/2$.

Proof. Form $\lfloor n/2 \rfloor \cdot \lceil n/2 \rceil$ pairs $\mathscr{P}$ of integers in the range from 1 to n by including each pair $\{i, j\}$ for which $i + j \equiv 1 \pmod 2$. For each pair $P \in \mathscr{P}$, form codewords by placing symbol 0 in the two positions in P, and form $(n - 2)!/2$ codewords of this type by placing the entries of each even permutation on $\{1, \ldots, n - 2\}$ in the remaining cells *in the same order*. Codewords arising from two disjoint pairs in $\mathscr{P}$ have distance at least four. Those arising from the same pair have distance at least three. Finally, other pairs of codewords have distance at least three; two arise from the differing location of the 0, and at least one further difference arises from the fact that the permutations in the remaining positions are both even. □

There are two main upper bounds that we employ: the Johnson and Plotkin bounds. The latter holds for *any* code, regardless of the constant composition property. The proofs of both upper bounds are standard and omitted. After each bound, we give an example of a composition and distance for which equality is achieved.

Proposition 1.3 (*Johnson bound*).

$$A([n_1, n_2, \ldots, n_k], d) \leqslant \frac{n}{n_1} A([n_1 - 1, n_2, \ldots, n_k], d).$$

Corollary 1.4. $A([n_1, \ldots, n_k], n) = \lfloor n / \max\{n_i\} \rfloor$.

Proof. Suppose without loss of generality that n_1 is the largest part in the composition. We can obtain equality with shifts of the codeword

$$\overbrace{1\ldots1}^{n_1}\,\overbrace{2\ldots2}^{n_2}\ldots\ldots\overbrace{k\ldots k}^{n_k}$$

by n_1 positions at a time. □

Proposition 1.5 (*Plotkin bound*). *Any k-ary code of length n and minimum distance d has at most*

$$\frac{d}{d - n + n/k}$$

codewords, provided the denominator is positive. Equality occurs if and only if the bound is an integer multiple of k, no pair of codewords are at distance n, and every symbol occurs equally often in each column.

Corollary 1.6. $A(1^1 2^m, 2m) = 2m + 1$.

Proof. The Plotkin bound is $2m+2$, but equality is impossible due to the composition. We have $A(1^1 2^m, 2m) = 2m+1$ using all cyclic shifts of the codeword $1234\ldots mm\ldots432$. □

2. Codes from polynomials

In this section, we use polynomials over finite fields to construct CCCs (see [15] for definitions).

Theorem 2.1. *Let $q = p^r = km + 1$ be a prime power. Then*

$$A(1^1 k^m, q - k) \geqslant \frac{q(q-1)}{k}.$$

Proof. Take a generator α for $GF^*(q)$. Let $\mathscr{P} = \{(ax + b)^k : a, b \in GF(q),\ a \neq 0\}$ be a set of polynomials of degree k. For each $f(x) \in \mathscr{P}$, construct one codeword

$$f(\cdot) = (f(0), f(\alpha^0), f(\alpha^1), \ldots, f(\alpha^{q-2})).$$

We claim that $C = \{f(\cdot) : f(x) \in \mathscr{P}\}$ is the desired CCC with $q(q-1)/k$ codewords. To prove the claim, we need to verify the minimal distance, weight distribution, and number of codewords contained in C.

1. For any two distinct polynomials $f(x), g(x) \in \mathscr{P}$, $f(x) - g(x) = 0$ has at most k roots, as the degree of $f(x) - g(x)$ is at most k and the polynomials are over $GF(q)$. Then the distance of any two different codewords of C is at least $q - k$.
2. Since $k \mid q - 1$, $\sigma(x) = x^k$ is a homomorphism from $(GF^*(q), \cdot)$ to $(GF^*(q), \cdot)$ with kernel size k. If $\sigma(\cdot)$ is applied to any permutation of $GF(q)$, the resulting vector has the desired weight distribution. The linear function $h(x) = ax + b$ with $a \neq 0$ is a permutation polynomial over $GF(q)$. Thus any $f(\cdot)$ with $f(x) \in \mathscr{P}$ has the desired weight distribution.
3. For any $f(x) \in \mathscr{P}$,

$$f(x) = (ax + b)^k = \sum_{i=0}^{k} \binom{k}{i} a^i b^{k-i} x^i.$$

If two polynomials $f_1(x)=(a_1x+b_1)^k$ and $f_2(x)=(a_2x+b_2)^k$ result in the same vector $f_1(\cdot)=f_2(\cdot)$, then

$$a_1^k=a_2^k,$$

$$ka_1^{k-1}b_1=ka_2^{k-1}b_2,$$

$$ka_1b_1^{k-1}=ka_2b_2^{k-1},$$

$$b_1^k=b_2^k.$$

From the first and the last equations, $a_1=\omega^{j_1}a_2$ and $b_1=\omega^{j_2}b_2$, where $\omega\in GF^*(q)$ with order k and $0\leqslant j_1, j_2\leqslant k-1$. With the fact that $p\nmid k$, the second and the third equations show that $k\mid j_1-j_2$ and $k\mid j_2-j_1$, thus $j_1=j_2$. So $f_1(\cdot)=f_2(\cdot)$ if and only if $a_1=\omega^j a_2$ and $b_1=\omega^j b_2$ with $0\leqslant j\leqslant k-1$. Thus $|C|=q(q-1)/k$. □

Taking $q=2\cdot((q-1)/2)+1$ with $k=(q-1)/2$ yields the Jacobsthal matrix construction in [21]. The following construction is known to be optimal when $q\leqslant 9$.

Corollary 2.2.

$$A\left(\left[\frac{q-1}{2},\frac{q-1}{2},1\right],\frac{q+1}{2}\right)\geqslant 2q,$$

where $q=p^m$ is an odd prime power.

Let $\alpha\in F=GF(q^m)$ and $K=GF(q)$. The *trace* $\mathrm{Tr}_{F/K}(\alpha)$ of α over K is

$$\mathrm{Tr}_{F/K}(\alpha)=\alpha+\alpha^q+\cdots+\alpha^{q^{m-1}}.$$

Theorem 2.3. *Let q be a prime power and m be a positive integer. Then*

$$A([\overbrace{q^{m-1},\ldots,q^{m-1}}^{q}],q^m-q^{m-1})\geqslant q(q^m-1).$$

Proof. Let $F=GF(q^m)$ and $K=GF(q)$. Let $\mathscr{P}=\{\mathrm{Tr}_{F/K}(ax+b):a,b\in GF(q^m),\ a\neq 0\}$. According to the definition of trace, each element of $\mathscr{P}$ is a polynomial of degree q^{m-1}. For each $f(x)\in\mathscr{P}$, construct one codeword

$$f(\cdot)=(f(0),f(\alpha^0),f(\alpha^1),\ldots,f(\alpha^{q^m-2})).$$

We claim that $C=\{f(\cdot):f(x)\in\mathscr{P}\}$ is the desired CCC with $q(q^m-1)$ codewords.

To prove the claim, we need to verify the minimal distance, weight distribution, and number of codewords contained in C.

1. Since the degree of each polynomial in $\mathscr{P}$ is q^{m-1}, the minimal distance of C is q^m-q^{m-1}.
2. $\mathrm{Tr}_{F/L}(\cdot)$ is a homomorphism from $(GF(q^m),+)$ to $(GF(q),+)$. So the weight distribution is $[\overbrace{q^{m-1},\ldots,q^{m-1}}^{q}]$.
3. To check the size of C, we use a method similar to that in the proof of Theorem 2.1. For convenience, we use Tr instead of $\mathrm{Tr}_{F/K}$. Suppose that two functions from $\mathscr{P}$ $f_1(x)=\mathrm{Tr}(a_1x+b_1)$ and $f_2(x)=\mathrm{Tr}(a_2x+b_2)$ give the same codeword, $f_1(\cdot)=f_2(\cdot)$. Then for any $x\in F$, $\mathrm{Tr}(a_1x+b_1)=\mathrm{Tr}(a_2x+b_2)$. Take $x=0$, and then $x=1$, to get

$$\mathrm{Tr}(b_1)=\mathrm{Tr}(b_2)\quad\text{and}\quad\mathrm{Tr}(a_1)=\mathrm{Tr}(a_2).$$

By linearity of the trace function, for any $x \in F$, $\mathrm{Tr}(a_1x) = \mathrm{Tr}(a_2x)$. Hence $\mathrm{Tr}((a_1 - a_2)x) = 0$, i.e. $(a_1 - a_2)x \in Ker(\mathrm{Tr}(\cdot))$, the kernel of the trace function. Also $a_1 - a_2 \in Ker(\mathrm{Tr})$. Then $a_1 - a_2 = 0$. As a conclusion, $\mathrm{Tr}(a_1x + b_1) = \mathrm{Tr}(a_2x + b_2)$ implies that $a_1 = a_2$ and $\mathrm{Tr}(b_1 - b_2) = 0$. Then $|C| = q^m(q^m - 1)/(q^m - 1) = q(q^m - 1)$. □

Example 2.4. Let $q = 3^2$ and $m = 2$. Then $A([3, 3, 3], 6) \geqslant 24$, which is indeed optimal according to Table III of [21]. We also see this code in Example 4.2.

In principle, Theorems 2.1 and 2.3 can be applied to any PA constructed from permutation polynomials or a fractional linear transformation over finite fields (refer to [4]). Here are two examples.

Example 2.5. Take a $PA(10, 8)$ with 720 permutations from $PGL(2, 9)$ [4]. Using $k = 2$ in Theorem 2.1, $A(1^2 2^4, 6) \geqslant 360$, where weights of 1 come from 0 and ∞. Using $\mathrm{Tr}_{F/K}$ with $F = GF(9)$ and $K = GF(3)$, $A([3, 3, 3, 1], 4) \geqslant 240$, where weight 1 comes from ∞.

3. Codes from distance-preserving mappings

The paper [3] investigates mappings f from $\mathbb{Z}_2^n$ to $\mathscr{S}_n$ that "preserve" (do not decrease) Hamming distance. Here, we continue these ideas and consider applications to constant-composition codes. The set of r-subsets of a set S is denoted by $\binom{S}{r}$. A *generalized distance-preserving map* $GDPM(m, n, d, r; q)$ is a function

$$f : X^m \to \binom{\mathscr{S}_n}{r},$$

where $|X| = q$ and such that

(i) $f(x)$ is a $PA(n, d)$ for all $x \in X^m$, and
(ii) $\mathrm{dist}(u, v) \geqslant \mathrm{dist}(x, y)$ for all $u \in f(x)$ and $v \in f(y)$.

It is always assumed that $2 \leqslant q \leqslant n$ and $r \geqslant 1$. When $r = 1$, we can take $d = n$. The condition that $m \leqslant n$ is required in all cases. Another necessary condition is $q^m r \leqslant n!$.

GDPMs behave nicely with respect to concatenation. More precisely, suppose that f is a $GDPM(m_1, n_1, d_1, r; q)$ and g is a $GDPM(m_2, n_2, d_2, r; q)$. Define $f \diamond g$ on $X^{m_1+m_2}$ by $(f \diamond g)(x_1x_2) = \{u_iv_i : i = 1, \ldots, r\}$, whenever $x_1 \in X^{m_1}$, $x_2 \in X^{m_2}$, and $f(x_1) = \{u_1, \ldots, u_r\}$, $g(x_2) = \{v_1, \ldots, v_r\}$. This definition depends on some arbitrary ordering of the u_i and v_i. Such an ordering is implicitly assumed. A typical element in the range of $f \diamond g$ is viewed as a concatenation of permutations over appropriate sets. When these sets are disjoint, we obtain:

Lemma 3.1. *If f is a $GDPM(m_1, n_1, d_1, r; q)$ and g is a $GDPM(m_2, n_2, d_2, r; q)$, then $f \diamond g$ forms a $GDPM(m_1 + m_2, n_1 + n_2, d_1 + d_2, r; q)$.*

The word lengths in a GDPM can also be incremented by one.

Lemma 3.2. *Suppose that $q \leqslant n$. If there exists a $GDPM(m, n, d, r; q)$, then there exists a $GDPM(m+1, n+1, d, r; q)$.*

Proof. Suppose that f is the hypothesized GDPM, and, for convenience, $X = \{1, \ldots, q\}$. Define f' on X^{m+1} by $u' \in f'(xa)$ if and only if

$$u'(i) = \begin{cases} u(i) & \text{if } i < n + 1, \ \ u(i) \neq a, \\ n + 1 & \text{if } u(i) = a, \\ a & \text{if } i = n + 1 \end{cases}$$

for some $u \in f(x)$, where $x \in X^m$ and $a \in X$. Clearly, $f'(xa)$ is a $PA(n + 1, d)$. Since $f(x) = \{u_1, \ldots, u_r\}$ is a $PA(n, d)$, we know that if $u_1(i) = u_2(i) = a$, then u_1 and u_2 differ in d positions other than position i. So u_1' and u_2', defined as above, differ in at least d positions. On the other hand, if $u_1(i) = u_2(j) = a$ with $i \neq j$, then $u'_1(i) = n+1$ and $u'_2(j) = n+1$. So u'_1 and u'_2 differ both in positions i and j, and in at least $d-2$ other positions. Thus $f'(xa)$ is a $PA(n+1, d)$. Suppose now that $u' \in f(xa)$ and $v' \in f(y, b)$ arise from $u \in f(x)$ and $v \in f(y)$, respectively. If $a = b$,

$$\mathrm{dist}(u', v') \geqslant \mathrm{dist}(u, v) \geqslant \mathrm{dist}(x, y) = \mathrm{dist}(xa, ya).$$

If $a \neq b$,

$$\mathrm{dist}(u', v') \geqslant \mathrm{dist}(u, v) + 1 \geqslant \mathrm{dist}(x, y) + 1 = \mathrm{dist}(xa, yb).$$

Therefore, f' is the required GDPM. □

Define $A_q(m, d)$ to be the maximum size of a q-ary code of length m and distance d. Lower bounds on $A_2(m, d)$ have been studied extensively in the literature. See [16].

Theorem 3.3. *Suppose that there exist GDPM*$(m_i, n_i, d_i, r; q)$ *for* $i = 1, \ldots, k$. *Let* $m = \sum m_i$ *and* $d \leqslant \sum d_i$. *Then there exists a CCC with composition* $[n_1, \ldots, n_k]^*$ *and distance d of size* $r \cdot A_q(m, d)$.

Proof. Let $|X| = q$ and suppose that $f_i : X^{m_i} \to \binom{\mathscr{S}_{n_i}}{r}$ are the hypothesized GDPMs. Define the permutations in the range of f_i as acting on the symbols $\{1, \ldots, n_i\}$. Let C be any code of length m and distance d over the alphabet X. Define

$$C' = \bigcup_{x \in C} (f_1 \diamond \cdots \diamond f_k)(x).$$

We claim that C' is the required CCC. First, the symbols of each word of C' are, in some order, $1, \ldots, n_1, 1, \ldots, n_2, \ldots, 1, \ldots, n_k$. So C' has constant composition $[n_1, \ldots, n_k]^*$. There are $r|C|$ elements in C' since $|f_i(x)| = r$ for all i, x. Finally, suppose that $u, v \in C'$. If u and v result from the same $x \in C$, the distance between u and v is at least $\sum d_i \geqslant d$ by condition (i) of the GDPM. On the other hand, if u and v result from codewords $x \neq y$, then their distance is at least $\mathrm{dist}(x, y) \geqslant d$ from condition (ii) of the GDPM. □

In [3], it was observed that a DPM$(n, n; q)$ gives rise to a $PA(nk, d)$ of size $A_q(nk, d)$. Apart from allowing $m \neq n$, Theorem 3.3 essentially strengthens this conclusion in the sense that $1^{nk} \preccurlyeq k^n$ as compositions. (See part (1) of Lemma 1.1.) But if all n_i are equal and a PA is desired, we can in fact multiply the bound above by a substantial factor.

Theorem 3.4. *If there exists a GDPM*$(m, n, d, r; q)$ *then there exists a PA*(nk, dk) *of size* $r \cdot A_q(mk, dk) \cdot A(1^k, \lceil dk/n \rceil)$.

Proof. Suppose that f is the given GDPM and C is a q-ary code of length mk and distance dk. By Lemma 3.1,

$$C' = \bigcup_{x \in C} (\overbrace{f \diamond \cdots \diamond f}^{k})(x)$$

forms a $GDPM(mk, nk, dk, r; q)$. So C' is a $PA(nk, dk)$. Now, for each word in C', the k disjoint blocks of n symbols used can be permuted according to a $PA(k, \lceil dk/n \rceil)$. The distance between words resulting from different permutations of blocks is at least $n\lceil dk/n \rceil \geqslant dk$, since no symbols from distinct blocks can agree. □

In [3], a $GDPM(4, 4, 4, 1; 2)$ is presented.

Corollary 3.5. (1) $A(1^{4n}, 4d) \geqslant A_2(4n, 4d) A(1^n, d)$.
(2) $A(n^4, d) \geqslant A_2(4n, d)$.

Using a hill-climbing algorithm, we have found various GDPMs with small parameters.

Lemma 3.6. *There exists the following GDPMs*: $GDPM(6, 6, 6, 2; 2)$, $GDPM(6, 6, 3, 3; 2)$, $GDPM(6, 7, 7, 7; 2)$, *and* $GDPM(4, 5, 5, 1; 3)$.

Proof. For the first map, consider the partial map below and use the automorphism $x \mapsto x+3$ to create 2-subsets at distance 6. Also, $f(\mathbf{x}+(1,0,\dots,0))$ is a transposition on the last two entries of $f(\mathbf{x})$.

000000 ↦ 012345	010000 ↦ 012435	001000 ↦ 012534	011000 ↦ 013245
000100 ↦ 014325	010100 ↦ 013425	001100 ↦ 014532	011100 ↦ 014235
000010 ↦ 021345	010010 ↦ 021435	001010 ↦ 021534	011010 ↦ 023145
000110 ↦ 024315	010110 ↦ 023415	001110 ↦ 024531	011110 ↦ 024135
000001 ↦ 102345	010001 ↦ 102435	001001 ↦ 104532	011001 ↦ 105432
000101 ↦ 135042	010101 ↦ 132405	001101 ↦ 103542	011101 ↦ 135402
000011 ↦ 152034	010011 ↦ 120345	001011 ↦ 204531	011011 ↦ 250413
000111 ↦ 124305	010111 ↦ 120435	001111 ↦ 201534	011111 ↦ 123405

For the second map, we cycle the first three coordinates of the following images to obtain the required PAs of distance 3.

000000 ↦ 503214	100000 ↦ 520314	010000 ↦ 305214	110000 ↦ 250341
001000 ↦ 450312	101000 ↦ 520143	011000 ↦ 105243	111000 ↦ 502413
000100 ↦ 043215	100100 ↦ 042351	010100 ↦ 301254	110100 ↦ 023415
001100 ↦ 403512	101100 ↦ 024315	011100 ↦ 103245	111100 ↦ 210345
000010 ↦ 450132	100010 ↦ 502134	010010 ↦ 503421	110010 ↦ 025431
001010 ↦ 054132	101010 ↦ 520431	011010 ↦ 051432	111010 ↦ 502143
000110 ↦ 034251	100110 ↦ 203154	010110 ↦ 130425	110110 ↦ 230451
001110 ↦ 034152	101110 ↦ 420531	011110 ↦ 031452	111110 ↦ 021435
000001 ↦ 453210	100001 ↦ 524301	010001 ↦ 351240	110001 ↦ 125304
001001 ↦ 541302	101001 ↦ 425013	011001 ↦ 145023	111001 ↦ 152340
000101 ↦ 314520	100101 ↦ 243510	010101 ↦ 314205	110101 ↦ 312540
001101 ↦ 413502	101101 ↦ 241503	011101 ↦ 314025	111101 ↦ 231045
000011 ↦ 345120	100011 ↦ 452130	010011 ↦ 513420	110011 ↦ 521034
001011 ↦ 145032	101011 ↦ 542130	011011 ↦ 135042	111011 ↦ 521403
000111 ↦ 431502	100111 ↦ 432150	010111 ↦ 314250	110111 ↦ 213450
001111 ↦ 314052	101111 ↦ 124035	011111 ↦ 413025	111111 ↦ 241053

Next, we cycle all coordinates of the images below for the required PAs of distance 7.

000000 ↦ 2035146	100000 ↦ 5142360	010000 ↦ 6432105	110000 ↦ 2360154
001000 ↦ 6425310	101000 ↦ 3150624	011000 ↦ 0254316	111000 ↦ 1365240
000100 ↦ 5104623	100100 ↦ 0563142	010100 ↦ 4601523	110100 ↦ 1325460
001100 ↦ 3146025	101100 ↦ 6213450	011100 ↦ 4501623	111100 ↦ 4506123
000010 ↦ 0645213	100010 ↦ 0426513	010010 ↦ 4521036	110010 ↦ 4326150
001010 ↦ 1063425	101010 ↦ 0634215	011010 ↦ 1064532	111010 ↦ 6431052
000110 ↦ 1530462	100110 ↦ 6130542	010110 ↦ 3264105	110110 ↦ 6130452
001110 ↦ 6250341	101110 ↦ 4062513	011110 ↦ 0134652	111110 ↦ 3401256
000001 ↦ 0541362	100001 ↦ 0263514	010001 ↦ 6412035	110001 ↦ 6104235
001001 ↦ 2435106	101001 ↦ 0243516	011001 ↦ 4650123	111001 ↦ 4516023
000101 ↦ 3541620	100101 ↦ 6124305	010101 ↦ 0452361	110101 ↦ 6504123
001101 ↦ 5602134	101101 ↦ 0243156	011101 ↦ 1643520	111101 ↦ 2405361
000011 ↦ 3026541	100011 ↦ 2153604	010011 ↦ 1654230	110011 ↦ 0436512
001011 ↦ 5203416	101011 ↦ 0421635	011011 ↦ 4652103	111011 ↦ 4031652
000111 ↦ 6145302	100111 ↦ 3041526	010111 ↦ 6132045	110111 ↦ 5124036
001111 ↦ 4165302	101111 ↦ 6350241	011111 ↦ 1250346	111111 ↦ 2163450

The GDPM from ternary words is now given.

0000 ↦ 02431	1000 ↦ 20431	2000 ↦ 21043
0100 ↦ 10432	1100 ↦ 12430	2100 ↦ 31042
0200 ↦ 31024	1200 ↦ 30421	2200 ↦ 10423
0010 ↦ 02341	1010 ↦ 21340	2010 ↦ 20341
0110 ↦ 01234	1110 ↦ 30412	2110 ↦ 10342
0210 ↦ 10324	1210 ↦ 12340	2210 ↦ 40321
0020 ↦ 02134	1020 ↦ 32140	2020 ↦ 20143
0120 ↦ 30214	1120 ↦ 30142	2120 ↦ 40132
0220 ↦ 02143	1220 ↦ 30124	2220 ↦ 40123
0001 ↦ 03241	1001 ↦ 23401	2001 ↦ 23041
0101 ↦ 01432	1101 ↦ 13402	2101 ↦ 41032
0201 ↦ 03421	1201 ↦ 13420	2201 ↦ 41023
0011 ↦ 21304	1011 ↦ 23410	2011 ↦ 41302
0111 ↦ 03412	1111 ↦ 41230	2111 ↦ 13042
0211 ↦ 01324	1211 ↦ 41320	2211 ↦ 43012
0021 ↦ 23104	1021 ↦ 23140	2021 ↦ 42013
0121 ↦ 03142	1121 ↦ 34102	2121 ↦ 43102
0221 ↦ 03124	1221 ↦ 34120	2221 ↦ 43120
0002 ↦ 21403	1002 ↦ 32401	2002 ↦ 24031
0102 ↦ 04231	1102 ↦ 34201	2102 ↦ 14032
0202 ↦ 01423	1202 ↦ 34021	2202 ↦ 43021
0012 ↦ 02314	1012 ↦ 24310	2012 ↦ 42301
0112 ↦ 04312	1112 ↦ 14230	2112 ↦ 40312
0212 ↦ 04321	1212 ↦ 14320	2212 ↦ 14023
0022 ↦ 04213	1022 ↦ 24103	2022 ↦ 24013
0122 ↦ 04132	1122 ↦ 34210	2122 ↦ 40213
0222 ↦ 04123	1222 ↦ 14203	2222 ↦ 42103

□

Example 3.7. From [16], $A_2(24,7) \geqslant 2^{12}$, so using the $GDPM(4,4,4,1;2)$, $A(6^4,7) \geqslant 2^{12}$. Refining the composition, there is a $1^4 5^4$ code of the same distance and size. Using the $GDPM(6,6,6,2;2)$ above, $A(4^6,7) \geqslant 2^{13}$. By comparison, inflating a $PA(6,2)$ by 4 gives a code with the same composition and distance 8, but with smaller size 6!.

We now give a result that trades intra-distance and length for multiplicity.

Theorem 3.8. *If there is a* $GDPM(n,n,n,1;q)$ *and a* $PA(k,d)$ *of size* r, *then there is a* $GDPM(n+k,n+k,d,r;q)$ *for all* $k \leqslant n/(q-1)$.

Proof. Suppose that X is the alphabet $\mathbb{Z}_q$. Suppose that f is the given GDPM. Define $e : X^k \to \binom{E}{k}$, where $E = \{1,\dots,n+k\}$, by

$$e(x_1,\dots,x_k) = \{x_1, x_2+q, \dots, x_k+q(k-1)\}.$$

We find $|e(x) \cap e(y)| = k - \mathrm{dist}(x,y)$ for any $x, y \in X^k$. Now we extend f to $g : X^{n+k} \to \mathscr{S}_{n+k}$ by $g(x_1x_2) = y_1y_2$, where $x_1 \in X^k$, $x_2 \in X^n$, y_1 is any ordering of the points of $e(x_1)$, and y_2 is the permutation defined by $f(x_2)$ on the points of $\{1,\dots,n+k\}\backslash e(x_1)$, say in increasing order. Now permute the coordinates of y_1 according to the $PA(k,d)$ to obtain the desired multiplicity. It is easy to check that the resulting map is distance-preserving. □

4. Codes from resolvable designs

Let X be a set of size n and $\mathscr{B}$ a collection of nonempty subsets of X (*blocks*) whose sizes belong to $\mathscr{K}$. If t and λ are positive integers, the pair $(X,\mathscr{B})$ is a *t-wise balanced design* with *index* λ (or simply a *design*) if every

t-subset of X is contained in exactly λ blocks. From now on, we consider only $\lambda = 1$. Design $(X, \mathscr{B})$ is *resolvable* if $\mathscr{B}$ can be partitioned into partitions (*resolution classes*) of X. If in addition each resolution class contains the same number of blocks of each size, the design is *class-uniformly resolvable*. The block set $\mathscr{B}$ of a design with $t = \lambda = 1$ is itself a partition of X; in this trivial case every design is class-uniformly resolvable. Of much greater interest are such objects with $t \geqslant 2$. For $t = 2$, there is a large base of literature on resolvable designs and a growing interest in the class-uniform condition. Relatively little is known about t-designs for large t, and even less about resolvable designs.

When there is one block size, $\mathscr{K} = \{k\}$, the class-uniform condition is vacuous. Such a design is a *Steiner system* $S(t, k, n)$. For a $S(t, k, n)$ to be resolvable, we need $k \mid n$. An easy counting argument shows there are $\binom{n}{t}/\binom{k}{t}$ blocks and thus $\binom{n-1}{t-1}/\binom{k-1}{t-1}$ resolution classes.

If we relax to the condition that every t-subset of X is contained in *at most* one block, then the result $(X, \mathscr{B})$ is a *packing*. A packing is (*class-uniformly*) *resolvable* in the same sense that a design is. A packing with $\mathscr{K} = \{k\}$ is denoted by $S'(t, k, n)$.

Theorem 4.1. *Suppose that there is a resolvable $S'(t, k, n)$ with r resolution classes and a $PA(n/k, \lceil d/k \rceil)$ of size s, where $d \geqslant (k - t + 1)n/k$. Then there exists a* CCC$(k^{n/k}, d)$ *of size rs.*

Proof. Arbitrarily order the blocks of each resolution class, say with labels $1, \dots, n/k$. Multiply each class according to the hypothesized PA on the blocks. From each resulting class, form codewords as follows: if symbol i occurs in the block with label j, put symbol j in position i. The distance between codewords resulting from different resolution classes is $\geqslant (k - t + 1) \cdot n/k$, since distinct blocks of the packing meet in less than t points. The distance between codewords resulting from the same resolution class is at least $k \cdot \lceil d/k \rceil$ since the PA guarantees that at least d/k pairs of blocks are disjoint. □

Example 4.2. For q a prime power, consider the *affine plane* of order q, a resolvable $S(2, q, q^2)$. There also exists $PA(q, q - 1)$ of size $q(q - 1)$. So by Theorem 4.1, there exists a CCC with composition $[q^q]$ and distance $q(q - 1)$ of size $(q + 1)q(q - 1)$. For $q = 3$, the construction of codewords is illustrated below.

123 456 789 → 111222333 147	258 369 → 123123123
123 789 456 → 111333222 147	369 258 → 132132132
456 123 789 → 222111333 258	147 369 → 213213213
456 789 123 → 333111222 258	369 147 → 312312312
789 123 456 → 222333111 369	147 258 → 231231231
789 456 123 → 333222111 369	258 147 → 321321321
159 267 348 → 123312231 168	249 357 → 123231312
159 348 267 → 132213321 168	357 249 → 132321213
267 159 348 → 213321132 249	168 357 → 213132321
267 348 159 → 312231123 249	357 168 → 312123231
348 159 267 → 231123312 357	168 249 → 231312123
348 267 159 → 321132213 357	249 168 → 321213132

The following result generalizes Theorem 4.1 in the case where block sizes are not uniform. The proof is similar.

Theorem 4.3. *Suppose that there is a class-uniformly resolvable t-wise balanced design with r resolution classes (or a packing with the same parameters) such that each class has m_i blocks of size i for $t \leqslant i \leqslant k$. Suppose also that there are $PA(m_i, d_i)$ of size s_i for each i. Let $d \geqslant \min\{\sum i d_i, \sum (i - t + 1)m_i\}$. Then there exists a CCC with composition $t^{m_t} \cdots k^{m_k}$ and distance d of size $r \prod s_i$.*

Example 4.4. A CCC$([3, 2, 2, 2], 6)$ of size 18 can be formed from a class-uniformly resolvable 2-design on nine points with six resolution classes, each consisting of one block of size three and three blocks of size two. (See [7] for this example.) In each class, the blocks of size two can be permuted according to a maximum $PA(3, 3)$.

5. Codes from computer search

To contrast and complement the exhaustive search methods in [21] for ternary codes, we present some nonexhaustive techniques for finding constant composition codes with small parameters.

Any computational method for constant composition codes must have some way of generating possible codewords. Given the composition vector $[n_1, \dots, n_k]$ stored as a static array, one can generate all $n!/n_1!n_2!\cdots n_k!$ words with this composition recursively as follows. Given an n-set X, procedures to find the lexicographically first m-subset, say $\texttt{first}(X, m)$, and next m-subset following Y, say $\texttt{nextmsub}(X, Y, m)$, of X are implemented (see [17] for details). Then, a recursive function is invoked that at the deepest level returns a partition of X into sets $X_1, \dots, X_k$ of sizes $n_1, \dots, n_k$. The resulting partition is converted to a codeword by placing symbol i in the positions indexed by X_i, $i = 1, \dots, k$.

```
rec(i):
    if (i = =k) then report codeword
    else Y := first(X', n_i)
        X := X\Y,   rec(i + 1),   X := X ∪ Y
        Y := nextmsub(X', Y, n_i)
        X := X\Y,   rec(i + 1),   X := X ∪ Y

X := {1, ..., n}
rec(1)
```

Clique search: This technique involves simply building a graph $G([n_1, \dots, n_k], d)$ whose vertex set is all possible codewords, with an edge between two vertices if the distance between corresponding words is at least d. The paper [21] discusses exhaustive clique search of this graph to find ternary CCCs. Alternatively, a probabilistic clique-finding algorithm, such as the one found in [1], can be used to find an approximate maximum clique in $G([n_1, \dots, n_k], d)$. Since graph size is a constraint, this method works well for coarse compositions and large distance. Some improvements on the bounds in [21] are given below.

Proposition 5.1. *We have* $A([5, 3, 1], 3) = 72$, *and*

$$\begin{array}{ll} A([4,3,2],3) \geqslant 216, & A([6,3,1],3) \geqslant 116, \\ A([5,4,1],3) \geqslant 168, & A([4,4,2],3) \geqslant 532, \\ A([4,3,3],3) \geqslant 690, & A([5,3,2],3) \geqslant 327, \\ A([5,4,1],4) \geqslant 76, & A([5,2,2],4) \geqslant 49. \end{array}$$

Greedy search: In this method, we begin with an empty array, and while looping through all possible codewords, we add one if it has distance at least d from every member of the current code. If the number of codewords is small enough to permit several greedy runs, a fixed number of codewords can be erased and the ordering of possible codewords changed in subsequent runs. Alternatively, several greedy passes can be made while declining to check a randomly chosen proportion of codewords. In any case, it is not necessary to use memory (other than storing the current code), since the distance test can be embedded in the construction of all codewords.

We have applied greedy search to some larger ternary compositions ($n > 10$) and certain quaternary compositions.

Example 5.2. Using repeated greedy search, we found that $A([4, 4, 4, 4], 9) \geqslant 403$. This improves upon the lower bound of 5×12 from resolvable 2-(16, 4, 2) designs with $PA(4, 3)$ mentioned in Theorem 4.1.

Building by columns: At times, it may be fruitful to dualize the notion of constructing a code "one word at a time". Instead, we fix a target M of codewords and hill-climb to find n columns of length M with the requirement that the desired composition and minimum distance are achieved. Using considerations from the Plotkin bound, it is best to assume that the possible columns are "equitable" with respect to the alphabet; that is, the numbers of occurrences of symbols $i \neq j$ differ by at most one for every $i, j = 1, \dots, k$. For the composition requirement, we never consider a column if the current family of columns already has n_i occurrences of symbol i in the same position. For the distance requirement, we do not add a column if it causes more than $n - d$ agreements in some pair of rows. For some compositions, this method has the advantage of reducing the search space. On the other hand, the method fails unless exactly n columns are produced, and it requires an initial guess of M.

Example 5.3. An easy strengthening of the Plotkin bound, Proposition 1.5, states that the size b of a k-ary code with length $2k$ and distance $2k-1$ satisfies $\binom{b}{2} \geqslant 2k(b-k)$, or

$$b \leqslant 2k - \tfrac{1}{2}\left(\sqrt{8k+1} - 1\right).$$

The (floor of the) right side is known [19] to be achieved for $k \leqslant 9$. Now consider such codes with composition 2^k. Using the technique of building by columns, we have found that the bound above is met with equality for $k = 1, 2, 4, 5$, but that for $k = 3$ the bound (four codewords) cannot be met. When $8k+1$ is a perfect square (say $k = 3, 6$), equality is achieved only if every pair of codewords intersects. While we have found a CCC(2^6, 11) of size 8, it remains an interesting open question whether the upper bound of 9 can be met. Examples for $k = 4$ and 5 are given below.

$k=4$	$k=5$
	0421031234
01320132	1340224130
13031022	2233411040
20103123	3004132142
22311300	3112404203
33200211	4120340312
	4302013421

6. Refining the composition

Here, we present a general construction that, given a CCC C of length n and certain CCCs of lengths $n_1, \ldots, n_k$, yields a CCC with more codewords than C and with a refined composition. This method was used with some success in [4] to recursively construct PAs from constant weight binary codes. Before presenting the construction, we require the notion of a transversal packing.

Suppose that X is a set partitioned into subsets X_i, where $|X_i| = g_i$ for $i = 1, \ldots, k$. A *transversal packing* of *distance* δ and *type* $g_1 g_2 \cdots g_k$ is a collection T of k-subsets of X with $|A \cap X_i| = 1$ for each i and $A \in T$ and such that $|A \cap B| \leqslant k - \delta$ for every $A, B \in T$.

Certain well-known constructions for transversal packings with both large and small distances are used:

- $\delta = k$, $|T| = \min\{g_1, \ldots, g_k\}$
  take disjoint k-sets across the X_i;
- $\delta = k - 1$
  use mutually orthogonal latin squares;
- $\delta = 1$, $|T| = \prod_i g_i$
  take all possible k-sets across the X_i.

Theorem 6.1. *Let C be a* CCC$([n_1, \ldots, n_k], d)$. *In addition, for $i = 1, \ldots, k$, let C_i be a* CCC$([n_{1i}, \ldots, n_{l_i i}], d_i)$ *with $n_{1i} + \cdots + n_{l_i i} = n_i$ that can be written as a disjoint union $C_i = \cup_j C_i^{(j)}$ of* CCC$([n_{1i}, \ldots, n_{l_i i}], d_i')$. *Suppose that there are transversal packings T_j of distance δ and type $|C_1^{(j)}| \cdots |C_k^{(j)}|$ for each j.*

Let $d = d_1 + \cdots + d_k$ and suppose that the sum of any δ of the d_i' is at least d. Then there is a CCC$([n_{11}, n_{21}, \ldots, n_{l_k k}], d)$ *of size*

$$|C| \sum_{j \geqslant 1} |T_j|.$$

Proof. Given a codeword w of C, we place the code C_i on symbols $i_1, \ldots, i_{l_i}$ in the positions corresponding to symbol i of C. Fix j and consider the $C_i^{(j)}$ as a partition for the transversal packing T_j. Form concatenations (over i) of rows of

$C_i^{(j)}$ according to the k-subsets of T_j. Now, take the union of such words over all j and over each $w \in C$. The size and composition of the resulting code are as required. It remains to verify the distance. By the condition on $(k-t+1)$-wise sums of the d_i', it follows that the minimum distance between codewords resulting from the same j and w is at least d. By the fact that $d = d_1 + \cdots + d_k$, concatenations from different j but the same $w \in C$ have distance at least d. Finally, since the minimum distance in C is d, and the C_i are on disjoint sets of symbols, the distance between words arising from different $w \in C$ is also at least d. □

7. Cyclic codes

In this section, we introduce cyclic CCCs, defined as CCCs with automorphism group containing a cyclic subgroup of order equal to the code length, i.e., the code contains a codeword and all its cyclic shifts. We present two constructions based on cyclotomic classes and circulant weighing matrices, respectively.

Cyclic CCCs can be viewed as FH sequences, which have been extensively studied in the area of spread spectrum communications. In this section, a couple of optimal FH sequences are constructed with respect to the well-known Lempel–Greenberger bound [14]. On the other hand, all the known constructions for FH sequences provide nice cyclic CCCs.

Let $X = \{x(j)\}$ and $Y = \{y(j)\}$ be two sequences with length v over a given alphabet A. Their *Hamming correlation* is defined as

$$H_{X,Y}(\tau) = \sum_{j=0}^{v-1} h[x(j), y(j+\tau)], \quad 0 \leqslant \tau \leqslant v-1,$$

where $h[x, y] = 1$ if $x \neq y$, and 0 otherwise, and all the operations among indices are performed modulo v.

Let S be the set of all sequences of length v over a given alphabet A. For $X \in S$, let

$$H(X) = \max_{0<\tau<v} \{H_{X,X}(\tau)\}.$$

A sequence $X \in S$ is *optimal* if $H(X) \leqslant H(X')$ for all $X' \in S$.

Lemma 7.1 (*Lempel–Greenberger bound [14]*). *For every sequence $Y = \{y(j)\}$ of length v over an alphabet A of size $|A| = m$,*

$$H(X) \geqslant \frac{(v-b)(v+b-m)}{m(v-1)},$$

where b is the least nonnegative residue of v modulo m.

The following corollary makes the above bound easier to use.

Corollary 7.2 (*Fuji-Hara et al. [9]*). *Suppose $v = am + b$ with $0 \leqslant b \leqslant m-1$. Then*

$$H(X) \geqslant \begin{cases} a & \text{if } v \neq m, \\ 0 & \text{if } v = m. \end{cases}$$

7.1. Cyclic codes based on cyclotomic classes

In this section, we present a method based on cyclotomic classes to construct cyclic CCCs with length $p = ef + 1$, where p is a prime. This method also leads to some optimal FH sequences.

Let $p = ef + 1$ be an odd prime. The *cyclotomic classes* C_i in $GF(p)$, $0 \leqslant i \leqslant e-1$, are $C_i = \{\alpha^{i+te} : 0 \leqslant t \leqslant f-1\}$, where α is a primitive element of $GF(p)$. The *cyclotomic numbers* of order e are $(i, j) = |(C_i + 1) \cap C_j|$.

The following equations can be derived from the definition of cyclotomic numbers.

Lemma 7.3 (*Storer [20]*). *For any i and j,*

(1) $|(C_i + w) \cap C_j| = |(w^{-1}C_i + 1) \cap w^{-1}C_j|$, *and*
(2) *if* $w^{-1} \in C_h$, *then* $|(C_i + w) \cap C_j| = (i + h, j + h)$.

Let $v = (a_0, a_1, \ldots, a_{m-1})$ be a sequence on the alphabet $\mathbb{Z}_e$. Then $supp_v(t) = \{i : a_i = t, 0 \leqslant i \leqslant m-1\}$ is the *support* of the symbol $t \in \mathbb{Z}_e$ in sequence v.

Lemma 7.4. *Let $p = ef + 1$ be an odd prime with e even and $C_0, C_1, \ldots, C_{e-1}$ be its cyclotomic classes. Construct a cyclic sequence $v = (a_0, a_1, \ldots, a_{p-1})$ of length p on the alphabet $\mathbb{Z}_e$ according to*

$$supp_v(0) = C_{\sigma(0)} \cup \{0\} \quad \textit{and}$$

$$supp_v(i) = C_{\sigma(i)}, \quad 1 \leqslant i \leqslant e-1,$$

where $(\sigma(0), \sigma(1), \ldots, \sigma(e-1))$ is a permutation of $(0, 1, \ldots, e-1)$. Then the sequence v forms a cyclic CCC with

$$A(f^{e-1}(f+1)^1, p-d) \geqslant p,$$

where d is determined based on two different cases:

$$d = \begin{cases} \sum_{i=0}^{e-1} (i, i) + 1 & \textit{if } f \textit{ is odd}, \\ \sum_{i=0}^{e-1} (i, i) + 2 & \textit{if } f \textit{ is even}. \end{cases}$$

Proof. The distance between v and its wth cyclic shift is equal to

$$\sum_{i=0}^{e-1} |(C_{\sigma(i)} + w) \cap C_{\sigma(i)}| + |\{w\} \cap C_{\sigma(0)}| + |\{0\} \cap (C_{\sigma(0)} + w)|.$$

If $w^{-1} \in C_h$, then the sum of the first e terms is equal to

$$\sum_{i=0}^{e-1} (\sigma(i) + h, \sigma(i) + h) = \sum_{i=0}^{e-1} (i, i).$$

This sum is independent of the value of w. Then

$$d = \sum_{i=0}^{e-1} (i, i) + \max_{w} (|\{w\} \cap C_{\sigma(0)}| + |\{0\} \cap (C_{\sigma(0)} + w)|).$$

The maximal value of $|\{w\} \cap C_{\sigma(0)}| + |\{0\} \cap (C_{\sigma(0)} + w)|$ is determined by whether there exist w and $-w$ belonging to $C_{\sigma(0)}$. Let $w = \alpha^t$, then $-w = \alpha^{t+(p-1)/2}$. Both w and $-w$ belong to $C_{\sigma(0)}$ if and only if $(p-1)/2 \equiv 0 \pmod{e}$. If e is even, both w and $-w$ belong to $C_{\sigma(0)}$ if and only if $(p-1)/2 \equiv 0 \pmod{e}$; that is, f is even. If e is odd, then f has to be even. Then $(p-1)/2 \equiv 0 \pmod{e}$, and both w and $-w$ are in C_0. □

We now give a related construction with more cyclic codewords.

Lemma 7.5. *Let $p = ef + 1$ be an odd prime and $C_0, C_1, \ldots, C_{e-1}$ be its cyclotomic classes. For each $k = 0, 1, \ldots, e-1$, define the cyclic sequence v_k by*

$$supp_{v_k}(0) = C_k \cup \{0\},$$

$$supp_{v_k}(i) = C_{k+i}.$$

Then the set of all cyclic shifts of all v_k forms a CCC with

$$A(f^{e-1}(f+1)^1, p-d) \geqslant ep,$$

where d is given by

$$d=\begin{cases}\max\{\sum_{i=0}^{e-1}(i,i)+2,\ \sum_{i=0}^{e-1}(k+i,i)+1 \mid 1\leqslant k\leqslant e-1\} & \text{if } f \text{ is even},\\ \max\{\sum_{i=0}^{e-1}(i,i)+1,\ \sum_{i=0}^{e-1}(k+i,i)+2 \mid 1\leqslant k\leqslant e-1\} & \text{otherwise}.\end{cases}$$

Proof. Lemma 7.4 tells us the distance for any single cyclic codeword. We only need to check the distance between two different cyclic codewords. The distance between X_l and the wth cyclic shift of X_k is equal to the maximum of the expression

$$\sum_{i=0}^{e-1}|(C_{i+k}+w)\cap C_{l+i}|+|\{w\}\cap C_{e-l}|+|\{0\}\cap(C_{e-k}+w)|.$$

If $w^{-1}\in C_h$, then the first summation is equal to

$$\sum_{i=0}^{e-1}(i+h+k,i+h+l)=\sum_{i=0}^{e-1}(k-l+i,i).$$

This sum is independent of the value of w.

The maximal value of $|\{w\}\cap C_{e-l}|+|\{0\}\cap(C_{e-k}+w)|$ is determined by whether there exists a w such that $w\in C_{e-l}$ and $-w\in C_{e-k}$. Since w runs through all of $\mathbb{Z}_p$, the summation is at least 1. We need to check the conditions under which the summation is 2.

Without loss of generality, suppose that $w\in C_{e-l}$. Then $w=\alpha^{e-l+te}$ for some t. Notice that $-w=\alpha^{(p-1)/2+(k-l)}$. So $-w\in C_{e-k}$ if and only if $(p-1)/2+(k-l)\equiv 0\ (\mathrm{mod}\ e)$. Now $k-l$ can be any value except 0. Therefore, the summation is 2 except if $(p-1)/2\equiv 0\ (\mathrm{mod}\ e)$, i.e., f is even. $\square$

To calculate the actual distance, we need to evaluate $\sum_{i=0}^{e-1}(k+i,i)$ for each possible value of k. With the following lemma, the task is quite simple.

Lemma 7.6 (*Storer [20]*). (1) *For any integers m and n*, $(i+me,j+ne)=(i,j)$.

(2) $(i,j)=(e-i,j-i)$.

(3)

$$\sum_{j=0}^{e-1}(i,j)=f-\theta_i \quad \text{where } \theta_i=\begin{cases}1 & \text{if } f \text{ is even and } i=0,\\ 1 & \text{if } f \text{ is odd and } i=e/2,\\ 0 & \text{otherwise}.\end{cases}$$

(4)

$$\sum_{i=0}^{e-1}(i,j)=f-\eta_i \quad \text{where } \eta_i=\begin{cases}1 & \text{if } j=0,\\ 0 & \text{otherwise}.\end{cases}$$

From these basic properties of cyclotomic numbers, we derive the following formula.

Lemma 7.7. *Let* $p=ef+1$ *be an odd prime. Then*

$$\sum_{i=0}^{e-1}(k+i,i)=\begin{cases}f-1 & \text{if } k=0,\\ f & \text{otherwise}.\end{cases}$$

Proof. By part (7.6) of Lemma 7.6, we have $(k+i,i)=(e-k-i,-k)$ for any i. Thus

$$\sum_{i=0}^{e-1}(k+i,i)=\sum_{i=0}^{e-1}(e-k-i,-k)=\sum_{i=0}^{e-1}(i,-k).$$

This summation is $f-1$ only when $k=0$ by part (7.6) of Lemma 7.6. $\square$

Theorem 7.8. *Let $p = ef + 1$ be an odd prime.*

(1) *If f is even, then there exists a cyclic CCC with*

$$A(f^{e-1}(f+1)^1, p-(f+1)) \geqslant \frac{p(p-1)}{f}.$$

(2) *If f is odd, then there exist cyclic CCCs with*

$$A(f^{e-1}(f+1)^1, p-f) \geqslant p \quad \text{and}$$
$$A(f^{e-1}(f+1)^1, p-(f+2)) \geqslant \frac{p(p-1)}{f}.$$

Proof. This follows directly from Lemmas 7.5 and 7.7. □

In terms of optimal FH sequences, we claim the following.

Corollary 7.9. *For any odd prime $p = ef + 1$ with f odd. There exists an optimal FH sequence X with length p, alphabet size f and $H(X) = f$.*

Proof. This is a restatement of $A(f^{e-1}(f+1)^1, p-f)) \geqslant p$. The optimum follows from Lemma 7.1. □

Compare this with Theorem 2.1, which claims that

$$A(1^1 f^e, p-f) \geqslant \frac{p(p-1)}{f}.$$

If we change the symbol with weight 1 to another symbol, the distance will decrease by at least 1 and at most 2. Theorem 7.8 points out how to get better distance when p is an odd prime. Of course, in the construction of Lemma 7.5, if we assign $\{0\}$ with a special symbol ∞, then we recover the cyclic version of Theorem 2.1, in which the length is a prime. We omit the details here.

This method can produce more cyclic codewords with shorter distance. However, a simple formula as in Theorem 7.8 is difficult to obtain, and we need more information about the cyclotomic numbers. The cyclotomic numbers are known for almost all values of e less than 24 and in a few other cases. Here is an example. Let f be an odd integer. Let $p = 4f + 1$ be a prime and of form $p = x^2 + 4y^2$ with $x \equiv 1 \pmod 4$. There are at most five different cyclotomic numbers of order 4. They are

$$(0,0) = (2,2) = (2,0) = \frac{p-7+2x}{16},$$
$$(0,1) = (1,3) = (3,2) = \frac{p+1+2x-8y}{16},$$
$$(1,2) = (0,3) = (3,1) = \frac{p+1+2x+8y}{16},$$
$$(0,2) = \frac{p+1-6x}{16},$$
$$\text{others} = \frac{p-3-2x}{16}.$$

Example 7.10. The following cyclic CCC shows $A([10, 9, 9, 9], 26) \geqslant 4 \cdot 37$.

0012233030020311031312331202212110032
0123300101131022102023002313323221103
0230011212202133213130113020030332210
0301122323313200320201220131101003321

From one of these cyclic orbits, $A([10, 9, 9, 9], 28) \geqslant 37$, which is an optimal FH sequence. Taking two orbits, it follows that $A([10, 9, 9, 9], 27) \geqslant 2 \cdot 37$.

Assign symbols on cyclotomic classes by

$$(supp(0), supp(1), supp(2), supp(3)) = (C_{\sigma(0)}, C_{\sigma(1)}, C_{\sigma(2)}, C_{\sigma(3)}),$$

where $(\sigma(0), \sigma(1), \sigma(2), \sigma(3))$ is a permutation of $(0, 1, 2, 3)$. Support assignments from the 12 "even" permutations of size 4 (those differing from the identity $(0, 1, 2, 3)$ by an even number of transpositions) give $A(37, [10, 9, 9, 9], 24) \geqslant 12 \cdot 37$.

Cyclotomic classes and cyclotomic numbers provide an efficient construction for cyclic CCCs. The optimum with respect to the Lempel–Greenberger bound suggests that the result could be optimal with respect to CCC, or close to it. The following example provides evidence that the resulting cyclic CCC could be optimal.

Example 7.11. Let $p = 7 = 3 \cdot 2 + 1$ with $e = 3$ and $f = 2$. With a very simple calculation, we can construct a cyclic CCC as follows:

0021120
0102201
0210012

Thus $A([3, 2, 2], 4) \geqslant 21$. According to [21], $A([3, 2, 2], 4) = 21$. If we take all possible assignments for cyclotomic classes, $A([3, 2, 2], 3) \geqslant 42$, which is optimal [21]. With respect to the Lempel–Greenberger bound, none of these sequences is optimal.

In general, FH sequence families do not provide cyclic CCCs. The reason is that they may not have constant weight distributions. However, any single FH sequence provides a cyclic CCC. The following is a known optimal construction for FH sequences.

Theorem 7.12 (*Lempel and Greenberger [14]*). *For any $q = p^n$ with p a prime and any $1 \leqslant t \leqslant n$, there exists an optimal FH sequence of length $q - 1$ and alphabet size p^t. Therefore,*

$$A([\overbrace{p^{n-t}, \ldots, p^{n-t}}^{p^t}, p^t - 1], p^n - p^{n-t}) \geqslant p^n - 1.$$

Example 7.13. Let $q = 9$ with $p = 3$, $n = 2$ and $k = 1$. Then $A([3, 3, 2], 6) \geqslant 8$. Again, this is best possible according to [21].

Recently there are some new optimal constructions for FH sequences [9].

7.2. Cyclic codes from circulant weighing matrices

A *circulant weighing matrix W* of order n and weight k, denoted by $CW(n, k)$, is a square matrix with entries from $\{0, 1, -1\}$ determined by its first row, and any other row being a cyclic shift of its predecessor, satisfying the weighing property: $WW^{\mathrm{T}} = kI_n$.

Any circulant weighing matrix is a cyclic CCC with certain distance and weight distributions. In this section, we review some results in this area and transform them into cyclic CCCs. The main task here is to provide an efficient way to compute the distance.

The following properties are well-known results on circulant weighing matrices.

Lemma 7.14. *Let W be a $CW(n, k)$ matrix. Then*

(1) $k = s^2$ *for some integer* s; *and*
(2) $m_+ = \frac{1}{2}s(s + 1)$, $m_- = \frac{1}{2}s(s - 1)$, *where m_+ and m_- denote the weight of* 1 *and* -1, *respectively.*

Lemma 7.15. *Let W be a* $CW(n,k)$ *and* $supp(0)$ *be the support of symbol* 0. *If*

$$\max_{1\leqslant i\leqslant n-1} |(supp(0)+i)\cap supp(0)|\leqslant\lambda,$$

then there exists a cyclic CCC with

$$A\left([m_+, m_-, n-k], \frac{3n-2k-3\lambda}{2}\right)\geqslant n,$$

where m_+ *and* m_- *represent the weights of* 1 *and* -1, *respectively, and can be computed via Lemma* 7.14.

Proof. The weight distribution and total number of codewords are easy to determine. Let $\{a_n\}_{i=0}^{n-1}$ and $\{b_n\}_{i=0}^{n-1}$ be any two different rows in W. To compute the minimal Hamming distance, we notice that the inner product of $\{a_n\}_{i=0}^{n-1}, \{b_n\}_{i=0}^{n-1}\in\mathscr{C}$ can be expressed as $A-D$, where A is the total number of pairs of $(\pm1,\pm1)$ and D is the total number of pairs of $(\pm1,\mp1)$. Since $WW^{\mathrm{T}}=kI_n$, we have $A-D=0$. We also need to know the number of pairs $(0,\pm1)$ and $(\pm1,0)$. The condition in the lemma implies that such number is $2(n-k-\lambda)$. Thus the Hamming distance we are looking for is

$$D+2(n-k-\lambda)=\frac{n-(2n-2k-\lambda)}{2}+2(n-k-\lambda)=\frac{3n-2k-3\lambda}{2}. \qquad \square$$

Lemma 7.16. *Let q be a prime power. There exists a* $CW(q^2+q+1,q^2)$, *and*

$$A\left(\left[\frac{q(q+1)}{2},\frac{q(q-1)}{2},q+1\right],\frac{q^2+3q}{2}\right)\geqslant q^2+q+1.$$

Proof. The construction for $CW(q^2+q+1,q^2)$ is as follows [10]:

Let D be a cyclic planar difference set with parameters (q^2+q+1,q^2). Let

$$\phi(x)=\sum_{d\in D}x^d$$

be the Hall polynomial. Then

$$\phi(x)^2=\sum_{d\in D}x^{2d}+2\sum_{e\neq f\in D}x^{e+f}.$$

It is proved that $\phi(x)^2$ has coefficients of x^i 0, 1, 2, i.e., $2d\neq 2e$ unless $d=e$, $e+f\neq e'+f'$ unless $e=e'$ and $f=f'$, and $2d\neq e+f$ unless $d=e=f$. Take $J(x)=\sum_{i=0}^{q^2+q}x^i$, and take the coefficients of $\phi(x)-J(x)$ as the sequence of the first row of the circulant weighing matrix. Refer to [10] for details. We care about the position of 0 in the resulting sequence. It is the set $2D$, which is still a planar difference set. The conclusion follows from Lemma 7.15. $\square$

Example 7.17. For $q=2$, $A([3,3,1],5)\geqslant 7$, which is optimal [21].

Example 7.18. The following sequence is the first row of $CW(21,16)$

$$+\ +\ +\ +\ +\ -\ +\ 0\ +\ 0\ -\ +\ +\ -\ 0\ 0\ +\ -\ 0\ -\ -$$

Take all the cyclic shifts of this sequence. We get $A([10,6,5],14)\geqslant 21$.

8. Conclusions

A variety of methods can be employed to construct constant composition codes. We have explored connections with generalized weighing matrices and with frequency hopping sequences. We employed cyclotomy and resolvable designs as the bases for constructive methods. We also developed heuristic computational search techniques. We have established techniques for producing codes with constant composition (including permutation codes) from binary codes. Each of these is useful in the construction of specific CCCs [5]; however, the wide variation in parameters for CCCs appears to necessitate such a multi-pronged approach.

While number-theoretic and algebraic techniques appear well suited to construction when each symbol appears equally often, in the remaining cases techniques that are more powerful appear to include computer search and that of distance-preserving maps. In any event, the powerful connections with other better-studied classes of designs and codes open a number of avenues for further examination.

Acknowledgements

Research of the authors is supported by the Army Research Office under Grant number DAAD 19-01-1-0406. Thanks to Malcolm Greig, Manish Gupta, and Violet Syrotiuk for useful discussions concerning this research.

References

[1] R. Battiti, M. Protasi, Reactive local search for the maximum clique problem, Algorithmica 29 (2001) 610–637.
[2] A.E. Brouwer, J.B. Shearer, N.J.A. Sloane, S.D. Warren, A new table of constant weight codes, IEEE Trans. Inform. Theory 36 (1990) 1334–1380.
[3] J.-C. Chang, R.-J. Chen, T. Kløve, S.-C. Tsai, Distance-preserving mappings from binary vectors to permutations, IEEE Trans. Inform. Theory 49 (2003) 1054–1059.
[4] W. Chu, C.J. Colbourn, P. Dukes, Permutation codes for powerline communication, Des. Codes Cryptog. 32 (2004) 51–64.
[5] W. Chu, C.J. Colbourn, P. Dukes, Tables for constant composition codes, J. Combin. Math. Combin. Comput. 54 (2005) 57–65.
[6] C.J. Colbourn, T. Kløve, A.C.H. Ling, Permutation arrays for powerline communication and mutually orthogonal Latin squares, IEEE Trans. Inform. Theory 50 (2004) 1289–1291.
[7] P. Danziger, B. Stevens, Class-uniformly resolvable designs, J. Combin. Des. 9 (2001) 79–99.
[8] H.C. Ferreira, A.J.H. Vinck, Interference cancellation with permutation trellis codes, in: Proceedings of the 52nd IEEE Vehicular Technology Conference, IEEE Press, Los Alamitos CA, 2000, pp. 2401–2407.
[9] R. Fuji-Hara, Y. Miao, M. Mishima, Optimal frequency hopping sequences: a combinatorial approach, IEEE Trans. Inform. Theory 50 (2004) 2408–2420.
[10] A.V. Geramita, J. Seberry, Orthogonal Designs: Quadratic Forms and Hadamard Matrices, Marcel Dekker, New York, Basel, 1979.
[14] A. Lempel, H. Greenberger, Families of sequences with optimal correlation properties, IEEE Trans. Inform. Theory 20 (1974) 90–94.
[15] R. Lidl, H. Niederreiter, Finite Fields, Cambridge University Press, Cambridge, 1997.
[16] S. Litsyn, Tables of best known binary codes, in: V. Pless, W.C. Huffman (Eds.), Handbook of Coding Theory, North-Holland, Amsterdam, 1998.
[17] A. Nijenhuis, H. Wilf, Combinatorial Algorithms, Academic Press, New York, 1975.
[18] N. Pavlidou, A.J.H. Vinck, J. Yazdani, B. Honary, Power line communications: state of the art and future trends, IEEE Comm. Magazine (2003) 34–40.
[19] B. Stevens, E. Mendelsohn, Packing arrays, Theoret. Comput. Sci. 321 (2004) 125–148.
[20] T. Storer, Cyclotomy and Difference Sets, Markham Publishing Company, Chicago, 1967.
[21] M. Svanström, P.R.J. Östergard, G.T. Bogdanova, Bounds and constructions for ternary constant-composition codes, IEEE Trans. Inform. Theory 48 (2002) 101–111.